

A reflection on the implicitly restarted Arnoldi method for computing eigenvalues near a vertical line

Karl Meerbergen

Raf Vandebril

Report TW 565, May 2010



Katholieke Universiteit Leuven
Department of Computer Science

Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

A reflection on the implicitly restarted Arnoldi method for computing eigenvalues near a vertical line

*Karl Meerbergen
Raf Vandebril*

Report TW 565, May 2010

Department of Computer Science, K.U.Leuven

Abstract

In this article, we will study the link between a method for computing eigenvalues closest to the imaginary axis and the implicitly restarted Arnoldi method. The extension to eigenvalues closest to a vertical line is straightforward, by incorporating a shift. Without loss of generality we will restrict ourselves here to the imaginary axis.

In a recent publication, Meerbergen and Spence discussed a new approach for detecting purely imaginary eigenvalues corresponding to Hopf bifurcations. The proposed method is based on inverse iteration (inverse power method) on a Lyapunov like eigenvalue problem. A projection step was added that significantly reduces the computational overhead.

The same method can be used for computing eigenvalues of a matrix pencil near a vertical line in the complex plane. This method then appears to be equivalent with Sorensen's implicitly restarted Arnoldi method with a special choice of shifts.

Keywords : Lyapunov eigenvalue problem, Kronecker eigenvalue problem, eigenvalues closest to the imaginary axis, implicitly restarted Arnoldi

A reflection on the implicitly restarted Arnoldi method for computing eigenvalues near a vertical line[☆]

Karl Meerbergen

Dept. Computer Science, Celestijnenlaan 200A, 3000 Heverlee (Leuven), Belgium

Raf Vandebril

Dept. Computer Science, Celestijnenlaan 200A, 3000 Heverlee (Leuven), Belgium

Abstract

In this article, we will study the link between a method for computing eigenvalues closest to the imaginary axis and the implicitly restarted Arnoldi method. The extension to eigenvalues closest to a vertical line is straightforward, by incorporating a shift. Without loss of generality we will restrict ourselves here to the imaginary axis.

In a recent publication, Meerbergen and Spence discussed a new approach for detecting purely imaginary eigenvalues corresponding to Hopf bifurcations. The proposed method is based on inverse iteration (inverse power method) on a Lyapunov like eigenvalue problem. A projection step was added that significantly reduces the computational overhead.

The same method can be used for computing eigenvalues of a matrix pencil near a vertical line in the complex plane. This method then appears to be equivalent with Sorensen's implicitly restarted Arnoldi method with a special choice of shifts.

Key words: Lyapunov eigenvalue problem, Kronecker eigenvalue problem, eigenvalues closest to the imaginary axis, implicitly restarted Arnoldi

1. Introduction

In this article, we will study a method for computing eigenvalues of a large sparse generalized eigenvalue problems, closest to the imaginary axis. This problem is of interest e.g. in the study of stability of dynamical systems where one is interested in computing Hopf bifurcations. Computing the specific values for which Hopf bifurcations arise, results in large, sparse eigenvalue problems. From earlier work [1–4], we know that detecting eigenvalues near the imaginary axis is not always an easy task. The reason is that most eigenvalue solvers search for eigenvalues near a target point, called a shift. The extension to eigenvalues closest to a vertical line is straightforward, by incorporating a shift. Without loss of generality we will restrict ourselves here to the imaginary axis.

The proposed method is a specific form of a new algorithm proposed in [5], for computing eigenvalues of a two-parameter eigenvalue problem $(A + \alpha B)\mathbf{x} = i\beta M\mathbf{x}$, with A, B and M real and $i^2 = -1$. The desired α is the one closest to zero corresponding to a pair of purely imaginary eigenvalues $\pm i\beta$. This problem is transformed to a Kronecker eigenvalue problem for which the eigenvalues of interest can be computed via inverse iteration (also called the inverse power method). The inverse iteration method is not performed on the very large Kronecker problem, but on

[☆]The second author has a grant as "Postdoctoraal Onderzoeker" from the Fund for Scientific Research–Flanders (Belgium). This research was also partially supported by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office, Belgian Network DYSCO (Dynamical Systems, Control, and Optimization) and by the Research Council K.U.Leuven, CoE EF/05/006 Optimization in Engineering (OPTeC).

Email addresses: karl.meerbergen@cs.kuleuven.be (Karl Meerbergen), raf.vandebril@cs.kuleuven.be (Raf Vandebril)

URL: <http://people.cs.kuleuven.be/karl.meerbergen> (Karl Meerbergen), <http://people.cs.kuleuven.be/raf.vandebril> (Raf Vandebril)

the corresponding Lyapunov system of equations to keep the computational complexity under control. The resulting Lyapunov eigenvalue problem is solved via projection on a Krylov subspace (based on Arnoldi) and then an extra projection step is needed for storing the n^2 vector by $O(n)$ parameters.

Though the approach performs well in practice, the additional projection step complicates a theoretical study of the convergence behavior. Without the extra projection step for keeping the memory-cost bounded, however, this coincides with performing inverse iteration on the large Kronecker eigenvalue problem. In addition, inverse iteration, computes only one eigenvalue. Though not discussed in [5] the role of single vector iterations can easily be replaced by a more generic form of subspace iteration [6, 7].

In this article, we will consider the more specific setting $M = B$, which corresponds to computing eigenvalues near the imaginary axis, as we shall see later. As a result of this choice, we are able to prove the link with the “Implicitly Restarted Arnoldi Method (IRAM)” for a particular choice of shifts. Furthermore, we will see how the theoretical setting (the link with IRAM), as well as the algorithms, admit a natural extension towards subspace iteration.

The Arnoldi procedure [8] is a well-known iterative manner for projecting a large arbitrary matrix via orthonormal vectors onto a smaller Hessenberg matrix. This Hessenberg matrix can then be used for approximating systems of equations [9–11], for approximating the spectrum of the original matrix [7, 8, 12], or for other large matrix computations. Especially relevant for our case, is solving the algebraic Lyapunov equation [13, 14].

One of the problems that might occur when using the Arnoldi procedure, is that the number of orthogonal Krylov vectors needed for a sufficiently accurate solution, can be rather large. To overcome this problem, restarting techniques are used where one stops expanding the current Krylov subspace, and uses the gained information for restarting with a new starting vector [15]. The most popular technique is Sorensen’s implicitly restarted Arnoldi method (IRAM) [16–18]. It does not restart explicitly with a single vector, but reduces the existing Krylov subspace to one of smaller dimension by performing steps of the QR-method. This technique allows performing QR-steps with shifts, such that one can emphasize or remove certain directions when computing the reduced Krylov space: the dimension k Krylov space is reduced to a dimension p Krylov subspace. Important for this paper is the connection with subspace iteration [19]. When the shifts are eigenvalue estimates (so-called Ritz values) from the Arnoldi method, the shifts are called exact. There is a nice connection with deflation of eigenvalues [17]. In that case, the reduced subspace is spanned by the p selected Ritz vectors. Further expansion of the reduced subspace can be interpreted as subspace iteration applied to the subspace of p kept Ritz vectors. This interpretation will appear to be helpful to understand the method in [5], in particular the case of $B = M$ that we consider here. The choice of p , i.e. the number of vectors in subspace iteration, is often chosen larger than the number of wanted eigenvalues, since this increases the speed of convergence. This choice is also known as thick restarting [20].

For the problem discussed here, we have the special case $B = M$. This corresponds to computing generalized eigenvalues $\lambda = \alpha \pm i\beta$ closest to the imaginary axis. Additionally, subspace iteration is enabled, resulting in the ability of computing a larger set of interesting eigenvalues, whereas inverse iteration only provides information related to one eigenvalue. It will be shown that the projection step carried out to reduce the rank of the solution of the Lyapunov solver corresponds to shrinking the related Krylov subspace. This provides us a link with the implicitly restarted Arnoldi method. The results provided here are a first step towards a better understanding and a more general theoretical framework for studying the approach of [5].

The article is organized as follows. In Section 2 we will discuss the related Kronecker and Lyapunov eigenvalue problem. Section 3 discusses how to compute the desired eigenvalues based on inverse subspace iteration. In Section 4 the link between inverse subspace iteration and implicitly restarted Arnoldi is discussed. Section 5 provides some numerical experiments. The conclusions are given Section 6.

2. The problem setting and equivalent eigenvalue problems

Computing the desired eigenvalues is based on a transformation of the problem to equivalent eigenvalue problems. This section discusses the related Kronecker and Lyapunov eigenvalue problems.

Consider the generalized eigenvalue problem

$$A\mathbf{x} = \lambda B\mathbf{x}, \quad (1)$$

with $A, B \in \mathbb{R}^{n \times n}$ and A and B nonsingular, whose eigenvalues $\lambda_i = \alpha_i + i\beta_i$ ($1 \leq i \leq n$) closest to the imaginary axis are desired. Hence, among all λ_i we are interested in the ones with the smallest $|\alpha_i|$. Generically, inverse (subspace)

iteration on (1) converges to the smallest $|\lambda_i|$ in absolute value. Therefore, convergence to the smallest $|\alpha_i|$ is not guaranteed.

Transforming the generalized eigenvalue problem to a Kronecker eigenvalue problem, eliminating thereby β_i solves this problem. The generalized eigenvalue problem $A\mathbf{x} = (\alpha + i\beta)B\mathbf{x}$ can be written as a two-parameter eigenvalue problem, where both α and β are now considered as (eigenvalue) parameters. Hence, we are interested in the smallest $|\alpha_i|$ which is either real ($\beta_i = 0$) or corresponds to a pair of purely imaginary eigenvalues $\pm i\beta_i$.

We have the following relation with the Kronecker eigenvalue problem. (More detailed information on the Kronecker eigenvalue problem, based on the bi-alternate product, can, for example, be found in [21].)

Theorem 1. Take $A, B \in \mathbb{R}^{n \times n}$, consider the following two eigenvalue problems:

$$A\mathbf{x} = \lambda B\mathbf{x} \quad (2)$$

$$\frac{1}{2}(A \otimes B + B \otimes A)\mathbf{z} = \gamma(B \otimes B)\mathbf{z}. \quad (3)$$

For each real eigenvalue pair (λ, \mathbf{x}) of Equation (2), $\gamma = \lambda$ is an eigenvalue of (3) with eigenvector $\mathbf{z} = \mathbf{x} \otimes \mathbf{x}$. For each complex conjugate eigenvalue couple λ and $\bar{\lambda}$ ($\lambda = \alpha + i\beta$) of Equation (2), $\gamma = \alpha$ is a double eigenvalue of (3) with $\mathbf{z} = \mathbf{x} \otimes \bar{\mathbf{x}}$ and $\bar{\mathbf{z}}$ as eigenvectors.

Conversely, if γ is an eigenvalue of (3), then there are eigenvalues λ_1 and λ_2 from (2), with $2\gamma = \lambda_1 + \lambda_2$. Moreover, \mathbf{z} is a linear combination of $\mathbf{x} \otimes \mathbf{y}$ and $\mathbf{y} \otimes \mathbf{x}$, where $A\mathbf{x} = \lambda_1 B\mathbf{x}$ and $A\mathbf{y} = \lambda_2 B\mathbf{y}$.

PROOF. Due to the appealing nature of the proof, we reconsider some parts of it (a more general form can be found in [5]). We first prove the case of two complex conjugate eigenvalues. Since A, B are real, all complex eigenvalues appear in pairs. Consider the eigenpairs $(\alpha + i\beta, \mathbf{x})$ and $(\alpha - i\beta, \bar{\mathbf{x}})$,

$$A\mathbf{x} = (\alpha + i\beta)B\mathbf{x} \quad \text{and} \quad A\bar{\mathbf{x}} = (\alpha - i\beta)B\bar{\mathbf{x}},$$

implying

$$(A - \alpha B)\mathbf{x} = i\beta B\mathbf{x} \quad \text{and} \quad (A - \alpha B)\bar{\mathbf{x}} = -i\beta B\bar{\mathbf{x}}.$$

Based on these equations, we get that $(\alpha, \mathbf{x} \otimes \bar{\mathbf{x}})$ is an eigenpair of (3).

$$\begin{aligned} [(A - \alpha B) \otimes B + B \otimes (A - \alpha B)](\mathbf{x} \otimes \bar{\mathbf{x}}) &= (A - \alpha B)\mathbf{x} \otimes B\bar{\mathbf{x}} + B\mathbf{x} \otimes (A - \alpha B)\bar{\mathbf{x}} \\ &= (i\beta B\mathbf{x} \otimes B\bar{\mathbf{x}}) + (B\mathbf{x} \otimes (-i\beta B\bar{\mathbf{x}})) = 0. \end{aligned}$$

Similarly, we can prove that $(\alpha, \bar{\mathbf{x}} \otimes \mathbf{x})$ is also an eigenpair of (3). In the case of real λ , that is, $\lambda = \alpha$, we use $\beta = 0$ giving us $(A - \alpha B)\mathbf{x} = 0$ from which we can deduce that $(\lambda, \mathbf{x} \otimes \mathbf{x})$ is an eigenpair of (3).

To prove the other direction we will simplify the problem, by multiplying (3) with $B^{-1} \otimes B^{-1}$. We get equivalence of (3) with (where $B^{-1}A = QRQ^H$ is the Schur decomposition, Q unitary, R upper triangular):

$$\begin{aligned} \frac{1}{2}(B^{-1}A \otimes I + I \otimes B^{-1}A)\mathbf{z} &= \gamma\mathbf{z} \\ \frac{1}{2}(R \otimes I + I \otimes R)(Q^H \otimes Q^H)\mathbf{z} &= \gamma(Q^H \otimes Q^H)\mathbf{z}. \end{aligned}$$

The eigenvalues of $(R \otimes I + I \otimes R)$ equal all possible combinations $\lambda_i + \lambda_j$ with λ_i, λ_j eigenvalues of (2). Moreover, also the structure of the eigenvectors is a clear consequence of this factorization. ■

The Kronecker eigenvalue problem (3) is closely related to a so-called Lyapunov eigenvalue problem. Consider Z an $n \times n$ matrix, the $\text{vec}(\cdot)$ operator stacks all columns of the matrix Z under each other. We get the following equivalent problems with $\text{vec}(Z) = \mathbf{z}$:

$$\begin{aligned} (A \otimes B + B \otimes A)\text{vec}(Z) &= 2\gamma(B \otimes B)\text{vec}(Z), \\ BZA^T + AZB^T &= 2\gamma BZB^T. \end{aligned} \quad (4)$$

We will refer to the second problem as the Lyapunov eigenvalue problem, where we are interested in the γ 's and matrices Z satisfying Equation (4). We will call the matrices Z 'eigenmatrices'.

Strictly speaking, we should call (4) a Sylvester eigenvalue problem, as Z can be nonsymmetric. But since we can restrict ourselves to symmetric solutions, we call this equation the Lyapunov eigenvalue problem.

It is interesting to remark that the corresponding eigenmatrices Z have low rank. Based on Theorem 1 we get that the eigenvectors \mathbf{z} of the Lyapunov eigenvalue problem are of the form $\mathbf{z} = \xi_1 \mathbf{x} \otimes \mathbf{y} + \xi_2 \mathbf{y} \otimes \mathbf{x}$, with ξ_1, ξ_2 two parameters. This gives us $\text{vec}(Z) = \mathbf{z}$ with $Z = \xi_1 \mathbf{y} \mathbf{x}^T + \xi_2 \mathbf{x} \mathbf{y}^T$, which is of rank 2. In the case that the eigenvalue γ corresponds to a real λ , the associated eigenmatrix is symmetric namely $\mathbf{x} \mathbf{x}^T$. In the other case, the eigenvalues γ are double and have associated non parallel eigenvectors \mathbf{x} and \mathbf{y} . The eigenvectors span therefore an invariant subspace of dimension 2. Considering the Lyapunov eigenvalue problem, one can construct a symmetric eigenmatrix and a skew-symmetric eigenmatrix generating the dimension 2 subspace of eigenmatrices. Both of the eigenmatrices are of rank 2, the symmetric one equals $\mathbf{y} \mathbf{x}^T + \mathbf{x} \mathbf{y}^T$ and the skew-symmetric one equals $\mathbf{y} \mathbf{x}^T - \mathbf{x} \mathbf{y}^T$.

3. Inverse subspace iteration with projection

Since we are interested in the eigenvalues of (2), with α closest to zero, we can apply inverse iteration on the Kronecker eigenvalue problem. Via Theorem 1, this procedure results in an eigenvalue γ corresponding to the eigenvalue λ with smallest $|\alpha|$.

Given a random starting vector $\mathbf{y}_0 \in \mathbb{R}^{n^2}$, inverse iteration computes iteratively normalized vectors $\mathbf{y}_j = \tilde{\mathbf{y}}_j / \|\tilde{\mathbf{y}}_j\|_2$, where

$$\frac{1}{2}(A \otimes B + B \otimes A) \tilde{\mathbf{y}}_j = (B \otimes B) \mathbf{y}_{j-1} \quad \text{for } j \geq 1.$$

Under some mild conditions, which are normally satisfied by random starting vectors (see e.g. [7]), the vector \mathbf{y}_i converges to the eigenvector belonging to the eigenvalue of (3) closest to zero.

This section is organized as follows. We first review the inexact inverse iteration method from [5]. Inverse iteration for (4) requires the solution of a Lyapunov equation, as we shall see further. We have inexact inverse iteration since the Lyapunov equation is solved by an iterative method. For the full description of the method we refer to [5]. In order to reduce the computational cost, a projection was added to inexact inverse iteration. As an extension, we consider inverse subspace iteration. This method is conceptually similar, but becomes slightly more complicated in the Lyapunov setting. We now discuss the inverse iteration method for (4).

3.1. Inverse iteration

Practically, it is, computationally inconvenient to work with $n^2 \times n^2$ matrices and vectors of length n^2 since A and B are already assumed to be large. Moreover, we also know that the desired eigenvector needs to be a sum of at most two tensor-decomposable vectors and hence only needs $4n$ parameters instead of n^2 .

Translating the above inverse iteration procedure to the Lyapunov eigenvalue problem, we get the following.

$$\frac{1}{2}(B \tilde{Y}_j A^T + A \tilde{Y}_j B^T) = (B Y_{j-1} B^T) \quad \text{for } j \geq 1.$$

Take a starting matrix $Y_0 \in \mathbb{R}^{n \times n}$ (Y_j is a normalized version of \tilde{Y}_j). By Theorem 1, we know that, for each eigenvalue, there exists a symmetric eigenmatrix, hence we will assume the solution symmetric and only search for symmetric solutions of (4). Solving the Lyapunov equation is as expensive as solving the corresponding system in the Kronecker setting. Especially the storage of n^2 parameters for the matrix Y_i is too expensive since we know that the resulting eigenmatrix Y_i has rank at most two. To reduce the computational complexity, we will approximate each Y_i by a low rank matrix of specified rank r . Since Y_i is real symmetric, the best approximation for a given rank r is obtained by approximating Y_i using a partial eigendecomposition, consisting of the r dominant eigenvalues. The matrix Y_j is thus not stored as a dense $n \times n$ matrix but in factored form $Y_j = W_j D_j W_j^T$, where W_j has orthonormal columns and D_j is a diagonal matrix. The solution techniques we will use in this article, generate already factored solutions: $\tilde{Y}_j = W_j \tilde{D}_j W_j^T$. As normalization, we use $D_j = \tilde{D}_j / \|\tilde{D}_j\|_F$. Generically r is taken larger than 2, not to endanger or to slow down too much the convergence. Note that starting even with a rank one right-hand side Y_0 , may lead to a high rank Y_1 after truncation of the smallest eigenvalues of Y_1 . After a few iterations, when the iterates start converging

to an eigenmatrix, a good approximation by a low rank matrix is possible without much loss of precision, as the eigenmatrices have rank at most two.

The problem, as it is considered here, has large n . There exists a variety of iterative solvers for the Lyapunov problem which can be found for example in [13, 22–28] for Krylov based methods, [29] for ADI type methods, and [30] for the Smith method. Overviews are presented in [31, 32].

We will use the block Arnoldi method for reducing the dimensions of the Lyapunov equation. The block size is equal to the rank of Y_{j-1} . For obvious reasons, this limits the rank of the solution to the number of Krylov vectors. In order to limit the cost of successively solving Lyapunov equations with a right-hand side of large rank, we can even reduce further the rank of the solution by truncating the small eigenvalues of \tilde{Y}_j as we discussed before. This is not always possible, in general, but for many operators, low rank solutions can be expected [33, 34].

The following algorithm implements the inverse iteration method as depicted above. The solutions of the Lyapunov equation are denoted by \tilde{Y}_j , and their normalizations by Y_j .

Algorithm 1 (Inverse Iteration on the Lyapunov Equation).

1. Given $Y_0 = W_0 D_0 W_0^T$, W_0 a column vector with $\|W_0\|_2 = 1$ and $\|D_0\|_F = 1$. Set $j = 0$.
2. While not converged
 - (a) Solve \tilde{Y}_{j+1} in factored form $\tilde{Y}_{j+1} = W_{j+1} D_{j+1} W_{j+1}^T$ from

$$\frac{1}{2}(B\tilde{Y}_{j+1}A^T + A\tilde{Y}_{j+1}B^T) = BY_jB^T.$$

- (b) Normalize \tilde{Y}_{j+1} and store it in Y_{j+1} .
- (c) Increase j : $j = j + 1$.

To check whether the method has converged we first compute the corresponding approximate eigenvalue γ as

$$\gamma = -\frac{\text{trace}(D_j \tilde{A}_j D_j \tilde{B}_j^T + D_j \tilde{B}_j D_j \tilde{A}_j^T)}{\text{trace}(2D_j \tilde{B}_j D_j \tilde{B}_j)}, \quad (5)$$

where $\tilde{A}_j = W_j^T A W_j$ and $\tilde{B}_j = W_j^T B W_j$. The Equation (5) is based on the Rayleigh quotient for Kronecker products see e.g. [35]. To check for convergence, we compute the associated α_j , β_j and \mathbf{x}_j from the projected small system $W_j^T (A - \alpha_j B) W_j \mathbf{x}_j = \iota \beta_j W_j^T B W_j \mathbf{x}_j$ and check the residual norm

$$\|(A - \alpha_j B) W_j \mathbf{x}_j - \iota \beta_j B W_j \mathbf{x}_j\|_2.$$

Computing this residual norm is not a bottle-neck since it is much faster than solving the corresponding Lyapunov equation.

When running this algorithm, the rank of the successive iterates Y_j may be large. Restricting the rank to a certain threshold r is surely helpful in order to reduce the computational cost of the Lyapunov solver. From experiments, we found that r can be ten or larger, which makes it potentially impractical for real life applications.

3.2. Inverse iteration with projection

In [5], the Lyapunov eigenvalue problem (4) is projected on the Krylov space used by the Lyapunov solver. Let V_{j+1} denote the basis vectors obtained from the block Arnoldi method for computing \tilde{Y}_{j+1} in Algorithm 1. Define

$$\tilde{B}_{j+1} = V_{j+1}^T B V_{j+1}, \quad \tilde{A}_{j+1} = V_{j+1}^T A V_{j+1} \in \mathbb{R}^{k \times k} \quad (6)$$

then we solve the order k Lyapunov eigenvalue problem

$$\frac{1}{2}(\tilde{B}_{j+1} \tilde{Z}_{j+1} \tilde{A}_{j+1}^T + \tilde{A}_{j+1} \tilde{Z}_{j+1} \tilde{B}_{j+1}^T) = \tilde{\gamma}_{j+1} \tilde{B}_{j+1} \tilde{Z}_{j+1} \tilde{B}_{j+1}^T. \quad (7)$$

The eigenmatrices have at most rank two. The corresponding Ritz eigenmatrices for the large scale Lyapunov eigenvalue problem (4) also have rank two: $Z_{j+1} = V_{j+1} \tilde{Z}_{j+1} V_{j+1}^T$. The advantage of the projection step is twofold: faster convergence is expected than with inverse iteration and the right-hand sides have rank two (at most). The projected equation (7) can be solved by the QZ method, which has a complexity of the order k^6 , or if this would be too expensive, inverse iteration using the Bartels and Stewart [36] direct linear system solver.

Algorithm 2 (Inverse Iteration with Projection on the Lyapunov Equation).

1. Take $Z_0 = W_0 D_0 W_0^T$. Set $j = 0$.
2. While not converged
 - (a) Compute the k Krylov vectors, denoted by V_{j+1} , generated for solving

$$\frac{1}{2}(B\tilde{Y}_{j+1}A^T + A\tilde{Y}_{j+1}B^T) = BZ_jB^T. \quad (8)$$

- (b) Solve the projected Lyapunov eigenvalue problem (7) and let \tilde{Z}_{j+1} be the eigenmatrix associated with the eigenvalue nearest zero. In practice, we decompose $\tilde{Z}_{j+1} = \tilde{W}_{j+1}D_{j+1}\tilde{W}_{j+1}^T$ with $\tilde{W}_{j+1} \in \mathbb{R}^{k \times r}$ with $r = 1$ or $r = 2$.
- (c) Compute the Ritz eigenmatrix in factored form $Z_{j+1} = W_{j+1}D_{j+1}W_{j+1}^T$ with $W_{j+1} = V_{j+1}\tilde{V}_{j+1}$.
- (d) Increase j : $j = j + 1$

The stopping criterion of this algorithm is the same as for the previous algorithm without projection step. Note that the block Krylov method for solving the (8) does not require D_j . This fact will be used for the extension to subspace iteration.

3.3. Inverse subspace iteration

As discussed before, inverse iteration only enables convergence towards a single eigenvalue. Assume now for robustness that we want to compute several eigenvalues simultaneously. To achieve this, we need subspace iteration. For simplicity of notation, we will now change to the Kronecker formulation. Instead of iterating on a single vector \mathbf{y}_j , we will now iterate on several vectors at the same time. Given ℓ starting vectors $[\mathbf{y}_0^{(1)}, \dots, \mathbf{y}_0^{(\ell)}]$, where the superscript (i) denotes the i th vector. Subspace iteration is of the following form, we solve for the $[\tilde{\mathbf{y}}_j^{(1)}, \dots, \tilde{\mathbf{y}}_j^{(\ell)}]$,

$$\frac{1}{2}(A \otimes B + B \otimes A)[\tilde{\mathbf{y}}_{j+1}^{(1)}, \dots, \tilde{\mathbf{y}}_{j+1}^{(\ell)}] = [\mathbf{y}_j^{(1)}, \dots, \mathbf{y}_j^{(\ell)}] \quad \text{for } j \geq 0. \quad (9)$$

where the columns of $[\mathbf{y}_{j+1}^{(1)}, \dots, \mathbf{y}_{j+1}^{(\ell)}]$ are orthonormal and constructed by using for instance, Gram-Schmidt orthogonalization on the columns of $[\tilde{\mathbf{y}}_{j+1}^{(1)}, \dots, \tilde{\mathbf{y}}_{j+1}^{(\ell)}]$. Let us denote the matrices containing these vectors as bold capital letters: \mathbf{Y}_{j+1} and $\tilde{\mathbf{Y}}_{j+1}$.

Algorithm 3 (Subspace iteration on the Kronecker problem).

1. Given a starting matrix $\mathbf{Y}_0 \in \mathbb{R}^{n^2 \times \ell}$, with orthonormal columns. Set $j = 0$.
2. While not converged
 - (a) Solve $\tilde{\mathbf{Y}}_{j+1}$ from $1/2(A \otimes B + B \otimes A)\tilde{\mathbf{Y}}_{j+1} = \mathbf{Y}_j$.
 - (b) Orthonormalize the columns of $\tilde{\mathbf{Y}}_{j+1}$ to get \mathbf{Y}_{j+1} .
 - (c) Increase j : $j = j + 1$.

Depending on the interest, one can check convergence for a single eigenvector or for multiple eigenvectors. Testing convergence for multiple eigenvectors proceeds identically as for a single eigenvector.

The structure of the eigenvectors is not exploited here and also the fact that dimensions are squared, is computationally very inconvenient. To overcome this problem, we switch back to the Lyapunov setting. The algorithm is very similar to the previous one. The major difference is that ℓ Lyapunov equations need to be solved, one for each Ritz vector, and that the resulting Krylov spaces are added together in a new subspace, which is then used for the projection. In the projection phase, ℓ Ritz pairs are computed.

Let the ℓ Ritz eigenmatrices be $W_j^{(i)}D_j^{(i)}W_j^{(i)T}$ for $i = 1, \dots, \ell$. Instead of solving a Lyapunov equation for each eigenmatrix, we can solve one Lyapunov equation with right-hand side matrix $W_jD_jW_j^T$ where the columns of W_j span all columns of $W_j^{(1)}, \dots, W_j^{(\ell)}$. The Krylov space generated by the block Arnoldi method on W_j is the sum of the Krylov spaces started with $W_j^{(i)}$, $i = 1, \dots, \ell$. This produces one large block Krylov space for all right-hand sides

together. Note that the matrix D_j is not used by the block Arnoldi method and is irrelevant. At first sight, there is no benefit compared to solving all Lyapunov equations together. However, dependencies in the blocks may occur so that the Krylov blocksize may be reduced during the execution of the block Arnoldi method [22]. The ℓ eigenmatrices are computed from an order k Lyapunov eigenvalue problem. The starting vectors for the next iteration are extracted from those eigenmatrices.

The following algorithm presents this idea.

Algorithm 4 (Subspace iteration on the Lyapunov problem).

1. Take $Z_0 = W_0 D_0 W_0^T$, with $W_0 \in \mathbb{R}^{n \times 1}$. Set $j = 0$.
2. While not converged
 - (a) Compute V_{j+1} of dimension $n \times k$ from an iterative method (block Arnoldi for example) with starting block of vectors W_j . This is related to solving a Lyapunov equation with right-hand side $Z_j = W_j D_j W_j^T$.
 - (b) Compute ℓ eigenpairs $(\tilde{\gamma}_i, \tilde{Z}^{(i)})$ for $i = 1, \dots, \ell$ of the Lyapunov eigenvalue problem (7).
 - (c) Compute the associated Ritz eigenmatrices in factored form $Z_j^{(i)} = W_{j+1}^{(i)} \tilde{D}^{(i)} (W_{j+1}^{(i)})^T$ for $i = 1, \dots, \ell$.
 - (d) Compute $W_{j+1} \in \mathbb{R}^{n \times \ell}$ so that all columns of $W_{j+1}^{(i)}$, $j = 1, \dots, \ell$ are spanned by the columns of W_{j+1} .

Convergence of this method can be checked again for the dominant eigenvalues or for multiple eigenvalues at the same time. Note that the matrix D_j is not used in this algorithm. Also note that in the first iteration, we start with the rank one matrix Z_0 . After the first iteration, we keep ℓ Ritz pairs. Starting with a rank one matrix is required to make the connection with IRAM later.

4. The relation with implicitly restarted Arnoldi

In this section, we will briefly recapitulate the (implicitly restarted) Arnoldi method [16, 18] and then consider the link with the Lyapunov eigenvalue problem.

The Arnoldi procedure is well-known for generating a sequence of orthonormal vectors, such that the resulting projected matrix is of Hessenberg form. Let us briefly recapitulate the construction of the orthonormal vectors. Let S be an $n \times n$ matrix, \mathbf{v}_1 a starting vector. The Krylov space of dimension k , with starting vector \mathbf{v}_1 is defined as $\mathcal{K}_k(S, \mathbf{v}_1) = \text{span}\{\mathbf{v}_1, S\mathbf{v}_1, S^2\mathbf{v}_1, \dots, S^{k-1}\mathbf{v}_1\}$. The Arnoldi procedure iteratively generates an orthonormal basis for the iteratively growing Krylov subspaces.

In each iteration, $S\mathbf{v}_j$ is orthogonalized against the previously computed orthonormal vectors into \mathbf{v}_{j+1} , which can be expressed as follows, with $h_{i,j}$ being the Gram-Schmidt coefficients.

$$S\mathbf{v}_j - h_{1,j}\mathbf{v}_1 + \dots + h_{j,j}\mathbf{v}_j = h_{j+1,j}\mathbf{v}_{j+1},$$

which, for $j = 1, \dots, k$ (let $k < n$), can be rewritten in matrix language as

$$S V_k = V_k H_k + h_{k+1,k} \mathbf{v}_{k+1} \mathbf{e}_k^T, \quad (10)$$

where $V_k = [\mathbf{v}_1, \dots, \mathbf{v}_k]$, $H_k = [h_{i,j}]$ is a $k \times k$ proper upper Hessenberg matrix. Eq. (10) is called the recurrence relation, since it gives the relation between successive iteration vectors. It is also called an order k Arnoldi factorization.

Under some mild conditions, the eigenvalues of the Hessenberg matrix H_k (named Ritz-values) approximate the well-separated eigenvalues of the matrix S [15, 37]. Often, the dominant eigenvalues are well-separated eigenvalues.

Let $H_k \mathbf{z} = \theta \mathbf{z}$, then θ is called a Ritz value and $\mathbf{y} = V_k \mathbf{z}$ an associated Ritz vector. The residual $\mathbf{r} = S\mathbf{y} - \theta \mathbf{y}$ can cheaply be computed from (10) as $\mathbf{r} = h_{k+1,k} \mathbf{v}_{k+1} \mathbf{e}_k^T \mathbf{z}$ and the residual norm $\|\mathbf{r}\| = h_{k+1,k} |\mathbf{e}_k^T \mathbf{z}|$. The stopping criterion of the Arnoldi method (and IRAM) is usually based on the residual norm, i.e. the method is stopped when $\|\mathbf{r}\|$ is below a prescribed tolerance.

Unfortunately, it may happen that a large number of iterations is required, before an accurate solution can be obtained. Storing a large number of iteration vectors becomes prohibitive. This was the motivation for the implicitly restarted Arnoldi method, which is explained in the following section.

4.1. Implicitly restarted Arnoldi

In the implicitly restarted Arnoldi procedure, we do not start from scratch with a new starting vector, but we shrink the existing Krylov subspace to a smaller dimension by removing unwanted directions from the subspace. Globally, the implicitly restarted Arnoldi procedure shrinks and expands a Krylov subspace on every restart.

Reducing the dimension of the Krylov subspace from k to p is done by performing $k - p$ steps of the shifted QR-method on the Hessenberg matrix. The orthogonal transformation is applied to the Krylov vectors. After each QR-step, the trailing orthogonal vector is removed and hence the dimension of the subspace is reduced by one. An important property of this QR-based reduction procedure is that the remaining vectors still span a Krylov subspace, but one of smaller dimension. Hence a successive application of $k - p$ QR-steps reduces the number of vectors to be stored from $k + 1$ to $p + 1$. Moreover, one can put more weight on some directions in the Krylov subspaces by well chosen shifts for the QR-steps. Roughly speaking, one can state that one should take the shifts close to the eigenvalues one does not want to keep. Hence the remaining directions are enhanced. Unfortunately, it is not always clear which shifts should be chosen for the shifted QR-method, nor what size of p should be taken.

Implicitly restarted Arnoldi applied on a matrix S is of the following form.

Algorithm 5 (Implicitly restarted Arnoldi method (IRAM)).

1. Given a starting vector \mathbf{v}_1 .
2. Build the order k Arnoldi factorization, starting from \mathbf{v}_1 .
3. While not converged to the desired eigenvalues
 - (a) Select $k - p$ shifts ν_1, \dots, ν_p .
 - (b) Apply a QR-step for each of the shifts ν_i ($1 \leq i \leq p$) on the Hessenberg matrix H_k , and remove the trailing vector from the Krylov space after each QR-step.
 - (c) Expand the existing Krylov space of order p to a space of dimension k by $k - p$ Arnoldi steps.

Convergence of this method is tested by computing the Ritz-values and checking if their residual norms are smaller than a prescribed tolerance.

The details of the method and its derivation can be found in Sorensen's work [16, 18]. Many choices of shifts are possible. We will use a selection of Ritz-values as shifts, so-called exact shifts. Assume matrix H_k has Ritz-values $\theta_1, \dots, \theta_k$ and associated Ritz-vectors $\mathbf{y}_1, \dots, \mathbf{y}_k$. Assume we want to keep $\theta_1, \dots, \theta_p$ and the directions $\mathbf{y}_1, \dots, \mathbf{y}_p$. Applying $k - p$ QR-steps in the implicit method with the remaining Ritz-values $\theta_{p+1}, \dots, \theta_k$ as shifts, filters out these Ritz vectors and leave us with the Krylov subspace $\text{span}\{\mathbf{y}_1, \dots, \mathbf{y}_p\}$. Expanding now again this Krylov subspace by Arnoldi gives us a new subspace spanned by k vectors. Moreover, it is proved in [17] that all the following subspaces

$$\text{span}\{\mathbf{y}_1, \dots, \mathbf{y}_p, S\mathbf{y}_i, S^2\mathbf{y}_i, \dots, S^{k-p-1}\mathbf{y}_i\} \quad \text{for } 1 \leq i \leq p, \quad (11)$$

span the same Krylov subspace of dimension k . It was observed in [16, 17] that restarting Arnoldi is not necessarily slower than a full Arnoldi process when exact shifts are used. The reason is that the shifts that are close to eigenvalues 'deflate' those eigenvalues from the subspace and this speeds up the convergence.

Hence the impact of the implicitly restarted Arnoldi method is two-fold: firstly, it filters out all undesired directions and, secondly, it enhances the other directions by subspace iteration. Indeed, from (11), we see that the p power sequences $\mathbf{y}_i, S\mathbf{y}_i, S^2\mathbf{y}_i, \dots, S^{k-p-1}\mathbf{y}_i$ lie in the Krylov space.

Another choice of shift is a zero shift. In that case, the subspace dimension is also reduced by one, i.e. the order k Arnoldi factorization

$$SV_k - V_k H_k = h_{k+1,k} \mathbf{v}_{k+1} \mathbf{e}_k^T$$

is transformed to the order $k - 1$ Arnoldi factorization

$$SV_{k-1}^+ - V_{k-1}^+ H_{k-1}^+ = h_{k,k-1} \mathbf{v}_k^+ \mathbf{e}_{k-1}^T$$

where $\text{Range}(V_k^+) = \text{Range}(SV_k)$ with $V_k^+ = [V_{k-1}^+, \mathbf{v}_k]$. We can combine both exact shifts and a zero shift.

Theorem 2. *Given the order k Arnoldi factorization (10). Let (θ_j, \mathbf{y}_j) for $j = 1, \dots, k$ be the k Ritz pairs. Then applying $k - p$ exact shifts and one zero shift leads to an Arnoldi factorization of order $p - 1$. By performing $k - p + 1$ additional Arnoldi steps, we obtain the Arnoldi factorization (10) where the columns of V_k span*

$$\{S\mathbf{y}_1, \dots, S\mathbf{y}_p, S^2\mathbf{y}_i, \dots, S^{k-p}\mathbf{y}_i\} \quad \text{for any } i : 1 \leq i \leq p. \quad (12)$$

PROOF. From [16, 17], applying the exact shifts $\theta_{p+1}, \dots, \theta_k$ produces the order p Arnoldi factorization

$$S V_p^+ - V_p^+ H_p^+ = h_{p+1,p}^+ \mathbf{v}_{p+1}^+ \mathbf{e}_p^T$$

where $\text{Range}(V_p^+) = \text{Range}(\mathbf{y}_1, \dots, \mathbf{y}_p)$. With one additional shift zero, we obtain the Arnoldi factorization

$$S W_{p-1} - W_{p-1} G_{p-1} = g_{p,p-1} \mathbf{w}_p \mathbf{e}_{p-1}^T,$$

where $\text{Range}([W_{p-1}, \mathbf{w}_p]) = \text{Range}(S V_p^+)$ [38]. Performing $k - p + 1$ additional Arnoldi steps produces an order k Arnoldi factorization (10). Since all powers $S^j \mathbf{y}_i$, $i = 1, \dots, p$, $j = 1, \dots, k - p$ lie in the Krylov space, the columns of V_k span (12). ■

For the solution of generalized eigenvalue problems (1), the Arnoldi method is often applied to $S = A^{-1}B$ which favours the convergence of the eigenvalues near zero. Therefore it is assumed that all eigenvalues near the imaginary axis are sufficiently close to zero, which is usually the case. The eigenvalues θ of H_k are now approximations to eigenvalues of S . In order to find an eigenvalue of (1), we must compute λ as $\lambda = \theta^{-1}$. It is the connection with this method and Algorithm 4 that will be given in the next section.

Alternatively, a shift can be used as in shift-and-invert Arnoldi, i.e. the Arnoldi method applied to $S = (A - \sigma B)^{-1}B$, which is helpful in some situations [4]. We will give an example further.

4.2. The solution of Lyapunov equations

Assume we have the following Lyapunov equation to solve:

$$A Y B^T + B Y A^T = \mathbf{w} \mathbf{w}^T.$$

Multiplying on the left with A^{-1} and on the right with A^{-T} , we have

$$Y S^T + S Y = S \mathbf{w} (S \mathbf{w})^T. \quad (13)$$

This problem can be solved by Arnoldi's method [13] applied to S with starting vector $S \mathbf{w}$. Suppose that V_k is the matrix of corresponding Arnoldi vectors and H_k is the Hessenberg matrix. Now consider the order k Lyapunov equation

$$\tilde{Y} H_k^T + H_k \tilde{Y} = \mathbf{e}_1 \mathbf{e}_1^T \|S \mathbf{w}\|^2$$

which is small if k is small and can be solved by Bartels and Stewart [36]. We use $Y = V_k \tilde{Y} V_k^T$ as approximate solution for (13).

When the right-hand side of (13) has rank larger than one, a block Krylov subspace method can be used [22]. This is, however, not needed as we explain in the next section. As we shall see, we have a right hand side that is the basis of a Krylov space. The following lemma then becomes useful.

Lemma 3. *If the columns of $V_0 \in \mathbb{R}^{n \times \ell}$ are a Krylov basis for S , then the block Arnoldi method applied to S with starting vectors V_0 reduces to the Arnoldi method.*

PROOF. This is a well-known property, and can be shown as follows. Since V_0 form a Krylov basis, we have that

$$S V_0 - V_0 H = R$$

where R is a rank one matrix. In the first block Arnoldi step, we orthogonalize $S V_0$ against V_0 . The remaining vectors form a rank one matrix, which proves the lemma.

4.3. Inverse iteration on the Lyapunov equation and IRAM

We first consider the case of $\ell = 1$, i.e., inverse subspace iteration corresponds to inverse iteration. The idea behind Algorithm 4 is to compute the solution of the Lyapunov equation by a Krylov method. In the first iteration, the right-hand side has rank one, and so, we can use the method described in the last section. Then, we project A and B on the subspace obtained by the Krylov method. Instead of projecting A and B , we can also project $S = A^{-1}B$. This is cheap, since this projection is known: it is the Arnoldi Hessenberg matrix H_k . The Ritz values $\theta_1, \dots, \theta_k$ and Ritz vectors $\mathbf{y}_1, \dots, \mathbf{y}_k$ are computed from H_k . Let the Ritz value nearest zero be real, θ_1 , with associated Ritz vector \mathbf{y}_1 . The solution of the order k Lyapunov eigenvalue problem (7) is the pair $(\theta_1, \mathbf{y}_1 \mathbf{y}_1^T)$. In the next iteration, a Lyapunov equation with right-hand side $S \mathbf{y}_1 \mathbf{y}_1^T S^T$ needs to be solved. We again use a Krylov method, this time with starting vector $S \mathbf{y}_1$. Following Theorem 2, using the $k - 1$ exact shifts $\theta_2, \dots, \theta_k$ and one zero shift, we obtain the single vector $S \mathbf{y}_1$. Following Theorem 2, k additional Arnoldi steps produce an order k Arnoldi factorization with starting vector $S \mathbf{y}_1$.

We can repeat this process until the eigenvalue has converged. In fact, the solution of a Lyapunov equation is not required. The Krylov space is entirely determined by the computation of $S \mathbf{y}_1$ on each iteration. This corresponds to IRAM with shifts $\theta_2, \dots, \theta_k$ and 0.

When we keep ℓ vectors instead of one, we have a similar situation. The ℓ eigenvectors of the Lyapunov eigenvalue problem are again rank one or two matrices whose eigenvectors are spanned by the columns of the rank ℓ matrix $[\mathbf{y}_1, \dots, \mathbf{y}_\ell]$. In order to perform the inverse subspace iteration step, we first have to form $[S \mathbf{y}_1, \dots, S \mathbf{y}_\ell]$. This is again easy to achieve by implicit restarting with shifts $\theta_{\ell+1}, \dots, \theta_k$ and one additional zero shift. Performing a block Arnoldi method on $[S \mathbf{y}_1, \dots, S \mathbf{y}_\ell]$ is not required either, as $[S \mathbf{y}_1, \dots, S \mathbf{y}_\ell]$ form the basis of a Krylov space. Theorem 2 can be used again. Note that, implicitly, we solve the Lyapunov eigenvalue problem and produce the eigenmatrices $\mathbf{y}_j \mathbf{y}_i^T$ for $i, j = 1, \dots, \ell$. Note that the Lyapunov equation does not have to be solved in practice. This corresponds to IRAM with shifts $\theta_{\ell+1}, \dots, \theta_k$ and 0.

5. Numerical examples

5.1. The Olmstead model

The mathematical model represents the flow of a layer of viscoelastic fluid heated from below [39, 40]. The equations are

$$\begin{aligned} \frac{\partial u}{\partial t} &= (1 - C) \frac{\partial^2 v}{\partial X^2} + C \frac{\partial^2 u}{\partial X^2} + Ru - u^3 \\ B \frac{\partial v}{\partial t} &= u - v \end{aligned}$$

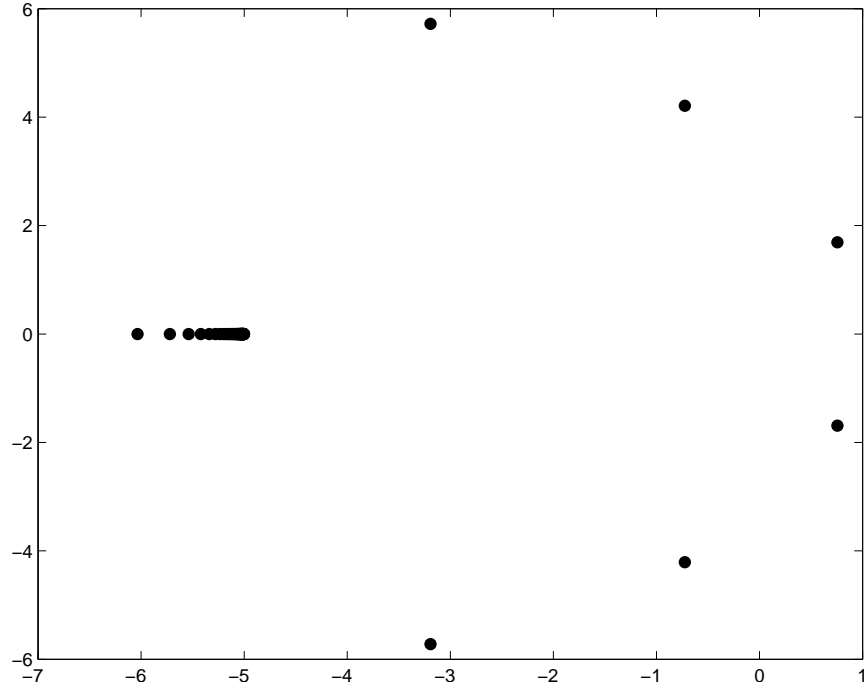
where u represents the speed of the fluid and v is related to viscoelastic forces. The boundary conditions are $u(0) = u(1) = 0$ and $v(0) = v(1) = 0$. After discretization with central differences with grid-size $h = 1/(n/2 + 1)$, the equations may be written as $d\mathbf{x}/dt = \mathbf{f}(\mathbf{x})$ with $\mathbf{x} = [u_1, v_1, u_2, v_2, \dots, u_{N/2}, v_{N/2}]^T$. We consider the Jacobian $A = \partial \mathbf{f} / \partial \mathbf{x}$ for $n = 10,000$, $B = 2$, $C = 0.1$ and $R = 0.6$, evaluated in the trivial steady state solution. Figure 1 shows the part of the spectrum near the imaginary axis. Most eigenvalues lie on the left of this picture far away from the imaginary axis.

As first selection (I) of shifts we used the $k - p$ eigenvalues of H_k that correspond with the left most Ritz values of (2), and as second selection (II), the $k - \ell = k - p - 1$ left most Ritz values and a zero shift. Choice (I) corresponds to the classical choice of (exact) shifts, where (II) corresponds to the method from [5]. Note that for both selections, the computational cost is of the same order for the same k and p . We see in Table 1 that with the second selection, the desired eigenvalue was found to full accuracy. The reason is that, with the first selection, the Arnoldi method first converged to the eigenvalues $0.75652 \pm 1.69189i$ and then started to converge to the desired eigenvalue. For the larger value of k , 20, we do not see significant different behaviour.

5.2. Purely imaginary eigenvalues

We generated an $n \times n$ matrix A with $n = 10,000$, $M = I$, such that A has eigenvalues $-1, -2, \dots, -9998$ and the complex pair $\pm 30i$. That means that the eigenvalues nearest the imaginary axis are the purely imaginary pair $\pm 30i$. This construction simulates the physical situation in the double-diffusive convection example [41, 42].

Figure 1: Part of the spectrum of the Olmstead equation



shifts	k	p	restarts	λ	residual norm
(I)	10	5	10	$-0.722241 \pm 4.20999i$	$8.3 \cdot 10^{-5}$
(II)	10	5	10	$-0.723921 \pm 4.20894i$	0
(I)	20	10	5	$-0.723921 \pm 4.20894i$	0
(II)	20	10	5	$-0.723921 \pm 4.20894i$	0

Table 1: Ritz values nearest the imaginary axis for different values of k and p and choices of shifts for the Olmstead problem

shifts	k	p	restarts	λ	residual norm
(I)	20	10	3	$+1.37 \cdot 10^{-14} \pm 30t$	$2.9 \cdot 10^{-12}$
(II)	20	10	3	$-2.04 \cdot 10^{-13} \pm 30t$	$9.7 \cdot 10^{-13}$
(I)	10	5	10	$+7.58 \cdot 10^{-14} \pm 30t$	$1.3 \cdot 10^{-10}$
(II)	10	5	10	$+3.21 \cdot 10^{-13} \pm 30t$	$2.4 \cdot 10^{-11}$

Table 2: Ritz values nearest the imaginary axis for different values of k and p and choices of shifts for the problem with purely imaginary eigenvalues

shifts	k	p	restarts	λ	residual norm
(I)	20	10	3	$+3.02 \cdot 10^{-14} \pm 30t$	$6.2 \cdot 10^{-16}$
(II)	20	10	3	$+3.55 \cdot 10^{-15} \pm 30t$	$7.1 \cdot 10^{-16}$
(I)	10	5	10	$-3.55 \cdot 10^{-15} \pm 30t$	$3.6 \cdot 10^{-18}$
(II)	10	5	10	$+2.49 \cdot 10^{-14} \pm 30t$	$1.2 \cdot 10^{-17}$

Table 3: Ritz values nearest the imaginary axis for different values of k and p and choices of shifts for the problem with purely imaginary eigenvalues, using $\sigma = 10$

From earlier work [1–4], we know that detecting eigenvalues nearest the imaginary axis is not always an easy task. The reason is that most eigenvalue solvers search for eigenvalues near a target point, called a shift.

In a first test, we compared the implicitly restarted Arnoldi method with $k = 20$ and $k = 10$ Krylov vectors and $p = 10$ and $p = 5$ vectors after the restart, respectively. As first selection (I) of shifts we used the $k - p$ eigenvalues of H_k that correspond with the left most Ritz values of (2), and as second selection (II), the $k - p - 1$ left most Ritz values and a zero shift. Note that for both selections, the computational cost is of the same order for the same k and p . Table 2 shows the Ritz values and their residual norms before each implicit restart for different values of k and p . We see that the convergence behaviour is very similar for both choices of shifts.

We now perform the same computation, but using a shift on A , i.e. we shift the matrix into $A - \sigma I$ with $\sigma = 10$. We then compute the eigenvalues nearest the vertical line through 10. Table 3 shows the results for the same parameters as the previous runs. Note that the residual norms are for the shift-and-invert transformation. As σ is now 10, a smaller residual norm does not necessarily imply a more accurate eigenvalue. However, we notice that the real part of the Ritz value has one more accurate digit with $\sigma = 10$.

5.3. Inverse subspace iteration for a parametrized eigenvalue problem

Recall the Olmstead equation from Section 5.1. We consider here the parametrized Jacobian $A + \alpha B$ for $n = 20,000$, $B = 2$, $C = 0.1$ with parameter $\alpha = R - 0.6$ where $R \in [0.6, 5]$, evaluated in the trivial steady state solution. In this example, we do not compute the eigenvalues nearest the imaginary axis, but we want to compute the values of R for which we have purely imaginary eigenvalues. That is, we want to compute α so that $i\beta$ is an eigenvalue of $(A + \alpha B)\mathbf{x} = i\beta M\mathbf{x}$.

We used Algorithm 4 with $\ell = 4$ and Krylov subspace dimension $k = 40$. Table 4 shows the computed α 's and the residual norms for four iterations of the method. Each line corresponds to an iteration (or restart) and each column to an eigenvalue. Each of the printed eigenvalues α have multiplicity two. The double eigenvalues have the same residual norms, so we do not print four columns.

α	residual	α	residual
0.600251	$1.8 \cdot 10^{-1}$	0.600251	$1.3 \cdot 10^0$
$0.2788 + 0.4861i$	$8.1 \cdot 10^{-3}$	$0.2788 - 0.4861i$	$6.3 \cdot 10^{-1}$
1.44783	$8.5 \cdot 10^{-7}$	-1.51304	$5.2 \cdot 10^{-4}$
1.44783	$1.9 \cdot 10^{-10}$	-1.51304	$9.8 \cdot 10^{-4}$

Table 4: Convergence behaviour for the computation of four eigenvalues of the parametrized Olmstead problem

6. Conclusions

In this article, an alternative approach, based on the Kronecker and Lyapunov setting was proposed for computing generalized eigenvalues closest to the imaginary axis. The method was based on inverse subspace iteration on the Lyapunov system, with an extra projection step to restrict the rank of the intermediate solutions. The link with implicitly restarted Arnoldi enabled us to theoretically predict the convergence of the method. Both methods perform subspace iteration on a selection of Ritz vectors obtained from a projection step.

The results in this article serve as a first step towards a better understanding of the more general method as proposed by Meerbergen and Spence in [5].

References

- [1] K. Cliffe, T. Garratt, A. Spence, Calculation of eigenvalues of the discretised Navier-Stokes and related equations, in: J. Whiteman (Ed.), *The mathematics of finite elements and applications VII MAFELAP*, Academic Press, 1990, pp. 479–486.
- [2] T. Garratt, G. Moore, A. Spence, Two methods for the numerical detection of Hopf bifurcations, in: R. Seydel, F. Schneider, H. Troger (Eds.), *Bifurcation and Chaos : Analysis, Algorithms, Applications*, Birkhäuser, 1991, pp. 119–123.
- [3] K. Cliffe, T. Garratt, A. Spence, Eigenvalues of block matrices arising from problems in fluid mechanics, *SIAM Journal on Matrix Analysis and Applications* 15 (1994) 1310–1318.
- [4] K. Meerbergen, D. Roose, Matrix transformations for computing rightmost eigenvalues of large sparse non-symmetric eigenvalue problems, *IMA Journal of Numerical Analysis* 16 (1996) 297–346.
- [5] K. Meerbergen, A. Spence, Shift-and-invert iteration for parameterized eigenvalue problems for the detection of Hopf bifurcations, *SIAM Journal on Matrix Analysis and Applications* To appear.
- [6] I. C. F. Ipsen, Computing an eigenvector with inverse iteration, *SIAM Review* 39 (2) (1997) 254–291.
- [7] D. S. Watkins, *The Matrix Eigenvalue Problem: GR and Krylov Subspace Methods*, SIAM, Philadelphia, Pennsylvania, USA, 2007.
- [8] Y. Saad, *Numerical Methods for Large Eigenvalue Problems*, Manchester University Press, Manchester, United Kingdom, 1992.
- [9] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd Edition, SIAM, Philadelphia, Pennsylvania, USA, 2003.
- [10] A. Greenbaum, *Iterative Methods for Solving Linear Systems*, SIAM, 1997.
- [11] G. H. Golub, C. F. Van Loan, *Matrix Computations*, 3rd Edition, Johns Hopkins University Press, Baltimore, Maryland, USA, 1996.
- [12] D. C. Sorensen, Numerical methods for large eigenvalue problems, *Acta Numerica* 11 (2002) 519–584.
- [13] Y. Saad, Numerical solution of large Lyapunov equations, in: M. A. Kaashoek, J. V. Schuppen, A. C. M. Ran (Eds.), *Signal Processing, Scattering, Operator Theory and Numerical Methods*, Birkhäuser, Boston, MA, 1990, pp. 503–511.
- [14] V. Simoncini, A new iterative method for solving large scale Lyapunov matrix equations, *SIAM Journal on Scientific Computing* 29 (3) (2008) 1268–1288.
- [15] C. Beattie, M. Embree, J. Rossi, Convergence of restarted Krylov subspaces to invariant subspaces, *SIAM Journal on Matrix Analysis and Applications* 25 (4) (2004) 1074–1109.
- [16] D. Sorensen, Implicit application of polynomial filters in a k-step Arnoldi method, *SIAM Journal on Matrix Analysis and Applications* 13 (1) (1992) 357–385.
- [17] R. B. Morgan, On restarting the Arnoldi method for large non-symmetric eigenvalue problems, *Mathematics of Computation* 65 (215) (1996) 1213–1230.
- [18] R. B. Lehoucq, D. C. Sorensen, C. Yang, *ARPACK Users Guide: Solution of Large Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods*, SIAM, 1998.
- [19] R. Lehoucq, On the convergence of an implicitly restarted Arnoldi method, *SIAM Journal on Matrix Analysis and Applications* 23 (1999) 551–562.
- [20] S. Stathopoulos, Y. Saad, K. Wu, Dynamic thick restarting of the Davidson, and the implicitly restarted Arnoldi methods, *SIAM Journal on Scientific Computing* 19 (1) (1998) 227–245.
- [21] W. Govaerts, Numerical methods for bifurcations of dynamical equilibria, SIAM, 2000.
- [22] M. Robb, M. Sadkane, Use of near-breakdowns in the block Arnoldi method for solving large Sylvester equations, *Applied Numerical Mathematics* 58 (4) (2008) 486–498.
- [23] V. Simoncini, A new iterative method for solving large-scale Lyapunov matrix equations, *SIAM Journal on Scientific Computing* 29 (3) (2008) 1268–1288.
- [24] V. Simoncini, V. Druskin, Convergence analysis of projection methods for the numerical solution of large Lyapunov equations, *SIAM Journal on Numerical Analysis* 47 (2) (2009) 828–843.
- [25] I. M. Jaimoukha, E. M. Kasenally, Krylov subspace methods for solving large Lyapunov equations, *SIAM Journal on Numerical Analysis* 31 (1) (1994) 227–251.
- [26] D. Y. Hu, L. Reichel, Krylov-subspace methods for the Sylvester equation, *Linear Algebra and its Applications* 172 (1992) 283–313.
- [27] A. S. Hodel, B. Tenison, K. R. Poolla, Numerical solution of the Lyapunov equation by approximate power iteration, *Linear Algebra and its Applications* 236 (1996) 205–230.
- [28] T. Gudmundsson, A. J. Laub, Approximate solution of large sparse Lyapunov equations, *IEEE Transactions on Automatic Control* 39 (5) (1994) 1110–1114.
- [29] R. Nong, D. C. Sorensen, A parameter free ADI-like method for the numerical solution of large scale Lyapunov equations, Tech. Rep. TR09-16, Rice University, CAAM, Houston, Texas (2009).
- [30] R. Smith, Matrix equation $xA + Bx = C$, *SIAM Journal on Applied Mathematics* 16 (1) (1968) 198–201.

- [31] S. Gugercin, J. Li, Smith-type methods for balanced truncation of large sparse systems, Springer Verlag, Berlin, Heidelberg, 2005.
- [32] V. Simoncini, Extended Krylov subspace for parameter dependent systems, *Applied Numerical Mathematics* 60 (5) (2010) 550–560.
- [33] T. Penzl, Eigenvalue decay bounds for solutions of Lyapunov equations: the symmetric case, *Systems and Control Letters* 40 (2000) 139–144.
- [34] A. C. Antoulas, D. C. Sorensen, Y. Zhou, On the decay rate of Hankel singular values and related issues, *Systems & Control Letters* 46 (2002) 323–342.
- [35] R. A. Horn, C. R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, Cambridge, 1991.
- [36] R. H. Bartels, G. W. Stewart, Solution of the matrix equation $AX + XB = C$, *Communications of the ACM* 15 (9) (1972) 820–826.
- [37] A. B. J. Kuijlaars, Which eigenvalues are found by the Lanczos method?, *SIAM Journal on Matrix Analysis and Applications* 22 (1) (2000) 306–321.
- [38] K. Meerbergen, A. Spence, Implicitly restarted Arnoldi with purification for the shift-invert transformation, *Mathematics of Computation* 66 (1997) 667–689.
- [39] W. Olmstead, W. Davis, S. Rosenblat, W. Kath, Bifurcation with memory, *SIAM J. Appl. Math.* 46 (2) (1986) 171–188.
- [40] M. Grinfeld, Dynamics of a model equation in viscoelasticity, in: *Proceedings Equadiff '91*, Vol. II, World Scientific, River Edge, NJ, 1993, pp. 568–572.
- [41] J. Turner, *Buoyancy effects in fluids*, Cambridge University Press, Cambridge, UK, 1973.
- [42] K. Cliffe, T. Garratt, A. Spence, Eigenvalues of the discretized Navier-Stokes equation with application to the detection of Hopf bifurcations, *Advances in Computational Mathematics* 1 (1993) 337–356.